Westminster International University in Tashkent

STUDENT USE		STAFF USE	
Module Name	Web Technology	First Marker's (acts as signature)	
Module Code	4BUIS011C	Second Marker's (acts as signature)	
Lecturer Name	Bunyod Khoshimkhujaev	Agreed Mark	
UoW Student IDs		**For Registrar's office use only (hard copy submission)**	
WIUT Student IDs	00007210		
Deadline date	15/11/2018		
Assignment Type	□Group✓Individual		
Word Count	507		

COURSEWORK SUBMISSION FORM

MARKERS FEEDBACK (Continued on the next page)

Contents

Introduction

The website I created for this coursework is a fast-food restaurant website. The design and content of the website is taken from HungryJacks official website. It has built in order to give information about the fast food chain, to build a positive brand image, and to promote the app of the website.

During the development process of the website, following tools and libraries were used:

- Twitter Bootstrap 4
- jQuery.js
- ekko-lightbox.js
- tempus-dominus-bootstrap-4.js
- moment.js
- bootstrap-select.js

Audience Profile

As the website is built for an Australian fast food franchise of Burger King, the main audience of the website is Australian people.

The website is a mean of attracting customers to the products, which is fast food in this case. In most cases, people who consume fast food are young singles between the ages of 20 and 34, students and families with young children who are looking for a quick meal. However, the website can also target women with its healthy food with reduced fat and sodium.

As the website is completely responsive and looks great in different devices. It can be used by all the target audience regardless of they use mobile, tablet or desktop.

Functionality and Features

One of the main functionalities of the website is that a user can order food directly from the website. Of course, it is not completely practical because of not using back-end programming and databases.

There is a food-ordering modal box with a form in it. It gets triggered when a user clicks on the “Order now” button on app.html file. The modal is built with the help of Twitter Bootstrap. The form includes 3 inputs, which are user’s name, type of food they want to order and time of delivery.

```
<div class="modal-content">
  <div class="modal-header">
    <h3 class="modal-title" id="orderTitle">ORDER AHEAD</h3>
    <button type="button" class="close" data-dismiss="modal" aria-label="Close">···
    </button>
  </div>
  <div class="modal-body">
    <form id="orderFood">
      <div class="form-group">
        <label for="userName">Your Name</label>
        <input type="text" class="form-control" id="usersName" name="usersName" placeholder="Enter your name here.." required>
      </div>
      <div class="form-group">
        <label class="my-1 mr-2" for="foodName">Place your order</label>
        <select class="selectpicker" id="foodName" data-live-search="true">
          <option data-tokens="ketchup mustard">HOT DOG, FRIES AND SODA</option>
          <option data-tokens="mustard">BURGER, SHAKE AND SMILE</option>
          <option data-tokens="frosting">HASH BROWN CHEESEBURGER SUPER STUNNER</option>
          <option>WHOPPER® HUNGER TAMERS</option>
          <option>GRILLED CHICKEN BACON & CHEESE HUNGER TAMERS</option>
          <option>TENDERCRISP® HUNGER TAMER</option>
          <option>BACON DELUXE™ HUNGER TAMER</option>
          <option>WHOPPER® FAMILY VALUE BUNDLE</option>
          <option>TENDERCRISP® FAMILY VALUE BUNDLE</option>
          <option>VEGAN CHEESEBURGER, FROZEN LIFT</option>
          <option>GRILLED CHICKEN WITH BACON & CHEESE, FROZEN FANTA® SOUR RASPBERRY</option>
          <option>BURGER, COLA CLASSIC AND FRIES</option>
        </select>
      </div>
      <div class="form-group">
        <div class="input-group date" id="orderTime" data-target-input="nearest">
          <input type="text" class="form-control datetimepicker-input" data-target="#orderTime" required>
          <div class="input-group-append" data-target="#orderTime" data-toggle="datetimepicker">
              <div class="input-group-text"><i class="far fa-clock"></i></div>
          </div>
        </div>
      </div>
    </form>
  </div>
  <div class="modal-footer">
```

The form uses bootstrap-select.js to make the food selection easier for user. They can choose from dropdown and even search for food names by typing it. The time

picker uses tempus-dominus-bootstrap-4.js to choose for user to pick a time for food delivery.

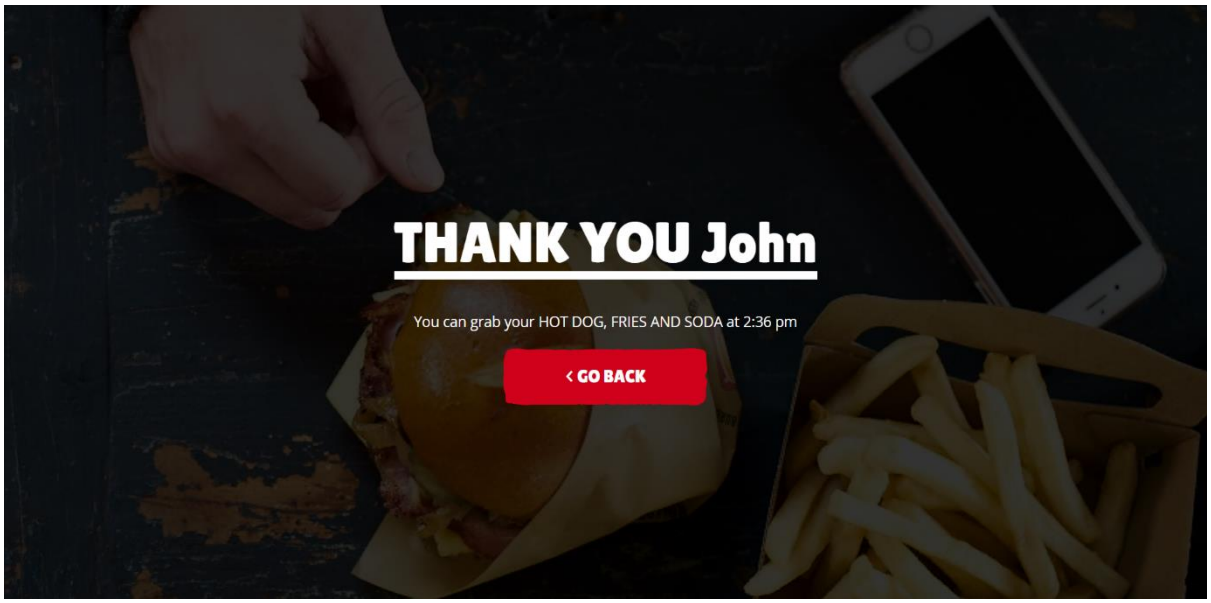
When user clicks “Order” button, the following function triggers:

```
// saving users input into local storage
var button = document.getElementById('submit');
var form = document.getElementById('orderFood');

var orders = []

button.addEventListener('click', (e) => {
  e.preventDefault()

  let usersName = document.getElementById('usersName').value || undefined
  let x = document.getElementById("foodName").selectedIndex;
  let food = document.getElementsByTagName("option")[x].value || undefined;
  let moment = $("#orderTime").datetimepicker('date') || undefined;
  let time = moment.format("h:mm a");

  let order = {
      name : usersName,
      food : food,
      time  : time
  }

  orders.push(order);


  // Stores the JavaScript object as a string
  localStorage.setItem('orders', JSON.stringify(orders));

  console.log(localStorage.getItem('orders'))

  form.reset()

  window.location.pathname = '/thankyou.html'

})
```

First, I selected submit button and the form itself by declaring variables for them. Then using AddEventListener, I added all the logic which gets called every time user clicks on the submit button. This function gets all the inputs of the user and creates an object called “order”. Then the “order” object gets pushed to the “orders” array. This array is stored in a local storage, which makes it possible to access the user’s inputs on another page. Lastly, the function redirects the user to thankyou.html file which also locates in the root. As redirection was made by

“window.location.pathname”, it only works if the website is accessed online or in a local server.

Once the above function gets executed, on the next page, I display the inputs that I have and let the user know that his or her order is successfully received.



In order to display this message, I need to access all the inputs of a user, which is stored in local storage. I did so with the following code:

```
window.onload = () => {
  let title = document.getElementById('title');
  let text = document.getElementById('text');
  let orders_string = localStorage.getItem('orders');
  let orders;

  orders = JSON.parse(orders_string);

  for (order of orders || []) {
    title.innerHTML = "THANK YOU " + order.name;
    text.innerHTML = "You can grab your " + order.food + " at " + order.time;
  }
}
```

The website hosted on the Internet using Github Pages. It is live at https://hungryjacks2018.github.io.